AF

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application No.	09/821,116
Filing Date	March 30, 2001
First Named Inventor	Michael N. Derr
Art Unit	2142
Examiner Name	Beatriz Prieto
Total Number of Pages in This Submission	
Attorney Docket Number	42390P10559

ENCLOSURES *(check all that apply)*

- ☒ Fee Transmittal Form
  - ☐ Fee Attached
- ☐ Amendment / Response
  - ☐ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
  - ☐ PTO/SB/08
- ☐ Certified Copy of Priority Document(s)
- ☐ Response to Missing Parts/ Incomplete Application
  - ☐ Basic Filing Fee
  - ☐ Declaration/POA
  - ☐ Response to Missing Parts under 37 CFR 1.52 or 1.53
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s)
  - ☐ Landscape Table on CD
- ☐ After Allowance Communication to TC
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☒ Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- ☐ Proprietary Information
- ☐ Status Letter
- ☒ Other Enclosure(s) *(please identify below):*

Return Post Card
Check $500.00

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm or Individual name	Paul A. Mendonsa, Reg. No. 42,879 BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
Signature	
Date	September 27, 2006

CERTIFICATE OF MAILING/TRANSMISSION

I hereby certify that this correspondence is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Typed or printed name	**Julie Dussault**		
Signature		Date	**September 27, 2006**

Based on PTO/SB/21 (09-04) as modified by Blakely, Sokoloff, Taylor & Zafman (wlr) 11/30/2005.
SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

FEE TRANSMITTAL for FY 2005

Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27.

TOTAL AMOUNT OF PAYMENT ($) 500.00

Complete if Known	
Application Number	09/821,116
Filing Date	March 30, 2001
First Named Inventor	Michael N. Derr
Examiner Name	Beatriz Prieto
Art Unit	2142
Attorney Docket No.	42390P10559

METHOD OF PAYMENT *(check all that apply)*

☐ Check ☐ Credit card ☐ Money Order ☒ None ☐ Other (please identify): _____

☒ Deposit Account Deposit Account Number: 02-2666 Deposit Account Name: Blakely, Sokoloff, Taylor & Zafman LLP

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☐ Charge fee(s) indicated below, except for the filing fee

☒ Charge any additional fee(s) or underpayment of fee(s) under 37 CFR §§ 1.16, 1.17, 1.18 and 1.20. ☒ Credit any overpayments

FEE CALCULATION

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet.	
2053	130	2053	130	Non-English specification	
1251	120	2251	60	Extension for reply within first month	
1252	450	2252	225	Extension for reply within second month	
1253	1,020	2253	510	Extension for reply within third month	
1254	1,590	2254	795	Extension for reply within fourth month	
1255	2,160	2255	1,080	Extension for reply within fifth month	
1401	500	2401	250	Notice of Appeal	
1402	500	2402	250	Filing a brief in support of an appeal	500.00
1403	1,000	2403	500	Request for oral hearing	
1451	1,510	2451	1,510	Petition to institute a public use proceeding	
1460	130	2460	130	Petitions to the Commissioner	
1807	50	1807	50	Processing fee under 37 CFR 1.17(q)	
1806	180	1806	180	Submission of Information Disclosure Stmt	
1809	790	1809	395	Filing a submission after final rejection (37 CFR § 1.129(a))	
1810	790	2810	395	For each additional invention to be examined (37 CFR § 1.129(b))	

Other fee (specify) _____

SUBTOTAL (2) ($) 500.00

SUBMITTED BY Complete (if applicable)

Name (Print/Type)	Paul A. Mendonsa	Registration No. (Attorney/Agent)	42,879	Telephone	(503) 439-8778
Signature				Date	09/27/06

Based on PTO/SB/17 (12-04) as modified by Blakely, Sokoloff, Taylor & Zafman (wlr) 12/15/2004.
SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	)	
	)	
Michael N. Derr for	)	
Intel Corporation	)	
	)	
Serial No.: 09/821,116	)	Group Art Unit: 2142
	)	
Filed: March 30, 2001	)	Examiner: PRITO, BEATRIZ

For: **Bit-Granular Writes of Control Registers**

APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Appellant submits this appeal brief, thus perfecting the notice of appeal filed on June 28, 2006. The required headings and subject matter follow.

I, Julie Dussault, hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail with sufficient postage in an envelope addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on September 27, 2006

Julie Dussault
(Signature of person mailing correspondence)

(i) Real party in interest.

This case is assigned of record to Intel Corporation, who is the real party in interest.

(ii) Related appeals and interferences.

There are no known related appeals and/or interferences.

(iii) Status of claims.

Claims 2, 3 and 5-25 are pending and stand rejected.

(iv) Status of amendments.

No amendments were filed subsequent to the final rejection.

(v) Summary of claimed subject matter.

Paragraph numbering of the filed application and the published application may differ. Accordingly, the following description references paragraphs of the present application based upon the paragraph numbering of the application as published. The supplied reference numbers and paragraphs are not meant to limit the scope of the present claims but merely to provide examples of some elements to aid understanding. The actual claim scope may be broader and/or narrower than the example elements given.

Claims 2, 3, 5-11 are dependent on independent claim 20, which relates to a method comprising receiving data of a write command comprising a bit enable field 401 and a data field 402, wherein each field comprises same number of bits (see figure – 4 and paragraphs [0033], [0034] and [0035]). The method further comprises

updating a register with one or more bits of the data field that are associated with enabled bits of the bit enable field (see paragraphs [0033], [0034] [0035] and [0037]).

Claim 2 relates to the method of claim 20, wherein the register is a control register for data transfer operation (see paragraphs 29 and 35).

Claim 5 relates to the method of claim 3, wherein the control register is command register (see paragraph 36).

Claim 6 relates to the method of claim 20, wherein some of the bits of said register are not overwritten (see paragraphs 32 and 34).

Claim 9 relates to the method of claim 20, wherein the data transfer operation comprises a data transfer between a processor subsystem 101 and an external storage device or peripheral such as, for example memory subsystem 106 and advanced graphics ports (AGP) 107 etc. (see figure 1, paras 18, 19, 21 and 37).

Claim 12 relates to a computer to receive a data value of a write directed to a control register and interpreting bits of the data value as a data field and bits of the data value as an enable field (see paragraphs [0033], [0034] and [0035]). The computer 100 further comprises overwriting only the bits at the bit locations of the control register for which the enable bit in the corresponding location in the bit enable field is set with the bit in the corresponding location in the data field (see figure – 4 and paragraphs [0033], [0034] and [0035]). Further, the computer comprises, a processor subsystem 101, a device 110 which transfers data to or from said processor subsystem; and a controller 103 connected between said device and said processor subsystem and adapted to control the transfer of data between said device and said processor subsystem (see figure 1, paragraphs 18, 22 and 23).

Claim 14 relates to the computer of claim 13, wherein the device comprises an IDE (integrated drive electronics) storage device and the bridge comprises an I/O controller hub (ICH) which controls an IDE data transfer between the processor subsystem and the IDE storage device (see paragraphs [0027]).

Claim 15 relates to the computer recited in claim 12, wherein the processor subsystem posts an entire command sequence in the controller for setting up the IDE (integrated drive electronics) data transfer (see paragraphs [0043]).

Claim 16 relates to a software program stored in tangible medium and when executed, causing a computer to execute a method comprising overwriting only the bits at the bit locations of the register for which a corresponding enable bit in the data value is set with corresponding data bits in the data value (see figure – 4 and paragraphs [0032], [0033], [0034] and [0035]). The method further comprises issuing a write of a data value to the register (see paras 29 and 30).

Claim 17 relates to a software program of claim 16, wherein said software program comprises a driver in the operating system software executed by a processor subsystem in the computer (see para 29 and 32).

Claim 18 relates to a software program of claim 17, wherein the register is a control register in a controller adapted to control an IDE (integrated drive electronics) data transfer operation between said processor subsystem and an IDE storage device (see para 32).

Claim 23 relates to a method of claim 20, wherein the control register has a location in I/O space and a location in memory space (see paras 30 and 35). The method further comprises to the processor subsystem issues a write of the data value to the location in memory space for the register (see paras 30 and 35).

Claim 25 relates to a method of claim 16, wherein the register has a location in configuration space and a location in memory space (see paras 30 and 35). The method further comprises issuing a write of a data value to the register comprises issuing the write to the location in memory space for the register (see paras 30 and 35).

(vi) Grounds of rejection to be reviewed on appeal

Whether claims 2, 3 and 5-25 are unpatentable under 35 U.S.C. §103(a) over Baker (US Patent No. 5,996,032) in view of Runaldue (US Patent No. 5,999,441).

(vii) Argument.

Claim Rejections under 35 USC 103 (Baker and Runaldue)

The rejection of Claims 2, 3 and 5-25 under 35 U.S.C. § 103(a), as being unpatentable over Baker (US Patent No. 5,996,032) in view of Runaldue (US Patent No. 5,999,441) is in error and should be reversed.

Claim 20

Claim 20 requires receiving data of a write command comprising a bit enable field and a data field and each field comprises same number of bits. In cited figure 10 and the related description, Baker teaches a register write circuitry 250 for writing an arbitrary number of data bits with a single write operation. In the register write circuitry 250, general purpose I/O (GPIO) write data input 252 is received by data flip flop 254. The data flip flop 254 also receives clock signal 256 and write enable input 258 from AND function 260. AND function 260 receives GPIO address okay signal 262, write strobe standard signal 264, and GPIO address bit input 266. AND function 260 provides the AND output 258 to write enable bit 268 of data flip flop

250. Only flip flop 250 with a 1 in particularly assigned address bit 266 will be written with the respective GPIO write data 252. Thus, Baker appears to disclose that the register write circuitry 250 permits writing only to bits that must change in a register, while preserving the previous value of the remainder of bits.

Appellant is unable to locate where Baker teaches receiving data of a write command comprising a bit enable field and a data field, wherein the bit enable field and the data field comprises same number of bits in each field. The Final Office Action appears to equate general purpose I/O (GPIO) write data input 252 of Baker with the write command of claim 20 of the Appellant. However, Appellant submits that Baker does not appear to teach that the general purpose I/O (GPIO) write data input 252 comprising a bit enable field and a data field and each field comprises same number of bits.

Baker further teaches that when the write enable is active, input write data 252 is written to flip flop 254 and appears on flip flop output 270. The Final Office Action appears to equate the flip flop output 270 with the write command of Appellant's claim 20. Appellant submits that Baker, in column 16, lines 64-67, discloses that address field 272 includes GPIO register address bits 274 and individual select field 276 address bits $A_0 - A_3$, according to the bit value, 0 or 1, of the associated bits in address field 272 (see figure 11 of Baker). Appellant submits that one skilled in the art would not consider flip-flop's output 270 similar to the write command of Appellant's claims 20.

In the office action the examiner has asserted that the features upon which applicant relies (i.e., sequential steps or acts, where a single write command is received and then a register is updated) were not recited in the rejected claims(s).

Appellant reiterate that claim 20 requires receiving data of a single command comprising a bit enabled field and data field and each field comprises equal number of bits and the register is updated with one or more bits of the data field that are associated with enable bits of the bit enable field. Appellant submits that these features are recited in claim 20.

Appellant further submits that Baker does not appear to teach receiving data of a write command wherein the data comprises a bit enable field and a data field comprising same number of bits and then updating a register with one or more bits of the data field that are associated with enable bits of the bit enable bits.

In addition, Runaldue appears to teach a bit enabled decoder logic 18 to enable bit by bit writing using data bits and mask bits, but there is no indication that the mask bits are part of the data bits transmitted in a write command. Further, Runaldue appears to be silent regarding how exactly the mask bits are transmitted. Runaldue thus does not teach receiving data of a single write command wherein the data comprises a bit enable field and a data field having same number of bits in each field as required by the Applicant's claim 20.

Appellant submit that the proposed combination of Baker and Runaldue does not teach receiving data of a write command comprising a bit enable field and a data field and updating a register with one or more bits of the data field comprises same number of bits and therefore the proposed combination does not arrive at the Appellant's invention of claims 20. Appellant respectfully requests the rejection of claim 20 be reversed.

Claims 2, 3 and 5

Claims 2, 3 and 5 include claim 20 as a base claim. Accordingly, claims 2, 3 and 5 are allowable for at least the reasons stated above in regard to claim 20. Appellant submits that the above submissions are sufficient to overcome the present rejection of claims 2, 3 and 5 under Baker and Runaldue. Applicant respectfully requests that the rejection of claims 2, 3 and 5 be reversed.

Claim 6

Claim 6 includes claim 20 as a base claim. Accordingly, claim 6 is allowable for at least the reasons stated above in regard to claim 20. Moreover, claim 6 requires some of the bits of said register are not overwritten.

In column 2, lines 25-32, Runaldue teaches that the logic circuit selectively causes a bistable latch storing to overwrite a stored value based upon the write signal and the corresponding bit enable signal. Hence a random memory can be fabricated, where word based addressing can be used to access a selected memory location while using a mask signal to write selected bits in the address memory word, without overwriting unselected bits of the addressed word.

The Final Office Action appears to equate, "without overwriting unselected bits of the addressed word", of Runaldue with "some of the bits of said register are not overwritten" of Appellant's claim 6. Appellant submits that one skilled in the art would not equate, "without overwriting unselected bits of the addressed word" of Runaldue with "not overwriting some of the bits of the register" of Appellant's claim 6, as the "unselected bits of the addressed word" of the Runaldue is not the same as the "bits of the register".

Further, Baker discloses a method where part of the address field is used to specify the bits to be loaded. However, Baker does not appear to teach some of the bits of said register are not overwritten as required by the Applicant's claim 6.

Appellant submit that the proposed combination of Baker and Runaldue does not teach some of the bits of said register are not overwritten as required by the Applicant's claim 6, and therefore the proposed combination does not arrive at the Appellant's invention of claim 6. Appellant respectfully requests the rejection of claim 6 be reversed.

Claims 7- 8

Claims 7-8 include claim 20 as a base claim. Accordingly, claims 7-8 are allowable for at least the reasons stated above in regard to claim 20. Moreover, claims 7 and 8 require data field and the bit enable field are received simultaneously and at respective address contiguous to each other.

In column 2, lines 14-25, Runaldue teaches that a static random access memory (SRAM) cell includes metal oxide semiconductor (MOS) transistor forming a bistable latch configured for storing a supplied data input. First and second supply transistors configured for selectively connecting the bistable latch to a voltage source in response to first and second gate signals, respectively. A logic circuit configured for generating the first and second gate signals in response to a write signal and bit enable signal. Thus Runaldue discloses the low level details of a random access memory (RAM) array which can be written at the bit level. Runaldue neither discloses anything of the bus level protocol (that is bit enable enclosed in the data phase) nor a system in which control register are written.

However, Appellant is unable to locate where Runaldue teaches, the data field and the bit enable field received simultaneously and at respective address contiguous to each other.

The Final Office Action appears to equate bistable latches of Runaldue with data field and the bit enable field received simultaneously of Appellant's claims 7-8. Appellant submits that the latches are needed to facilitate transfer of data, whereas the data field and the bit enabled field are to store respective data therein. Appellant submits that one skilled in the art would not equate bistable latch of Runaldue with data field and the bit enable field received simultaneously as required by Appellant's claims 7-8.

Also, Baker discloses a method where part of the address field is used to specify the bits to be loaded. However, Baker does not appear to teach that the data field and the bit enable field are received simultaneously and at respective address contiguous to each other, as required by the Applicant's claims 7-8.

Appellant submit that the proposed combination of Baker and Runaldue does not teach that the data field and the bit enable field are received simultaneously and at respective address contiguous to each other, as required by the Applicant's claims 7-8, and therefore the proposed combination does not arrive at the Appellant's invention of claims 7-8. Appellant respectfully requests the rejection of claims 7-8 be reversed.

Claim 9-11

Claims 9 -11 include claim 20 as a base claim. Accordingly, claims 9 -11 are allowable for at least the reasons stated above in regard to claim 20. Moreover, claim 10 require that the processor subsystem posts the entire command sequence

for setting up the data transfer. In Figure 2, Runaldue shows a circuit diagram illustrating in detail the SRAM cell according to an embodiment of his invention, however no component has been marked by a numeral 137 as cited by the Examiner in the Final Office Action. Further, Appellant is unable to locate numeral 137 in Runaldue. Appellant submits that Runaldue (column 4, lines 21-25) discloses that the SRAM cell includes a logic circuit configured for generating gaiting signals in response to write enable signal (WRTDAT) and a bit enable signal (BIT_EN), where the gaiting signal causes the SRAM cell to selectively overwrite a stored data value with the supplied data value. Thus, Runaldue discloses the low level details of a random access memory (RAM) array which can be written at the bit level. Runaldue neither discloses anything of the bus level protocol (that is bit enable enclosed in the data phase) nor a system in which control register are written.

In column 14/lines 64-66, Baker discloses that a check is then made to determine whether the command is a receive, transmit, PCI to/from local bus or auxiliary command. Further, in column 6/lines 39-59, Baker discloses that PCI bus logic 60 implements the logic for interfacing PCI interface ASIC 20 to PCI bus. PCI slave logic 66 provides the ability for external PCI agent to read and write slave interface control logic for assessing all of the PCI interface 20 and status registers 68, 76, 88 and 92 which are required by application software to control the operation of PCI – interface ASIC 20 and monitor its operational status.

However, Appellant is unable to locate where Runaldue or Baker teaches the processor subsystem posts entire command sequence for setting up the data transfer. Appellant submits that even the combination of Runaldue and Baker does

not anticipate the invention of the claim 10 of the Appellant. Appellant respectfully requests the rejection of claims 9-11 be reversed.

Claim 12

Claim 12 require receiving a data value of a write directed to a control register, interpreting bits of the data value as a data field and as enable bits in bit enable field and the number of bits being equal to the number of bits in the control register.

In its abstract, Baker discloses that register write circuitry 250 writes to plurality of data register bits 276 using a single register write operation by storing both designated address bits 276 and address bits 274 for addressing a predetermined data register. Applicant submits that there appear to be an error in the abstract, wherein 274 has been referred as data. Upon looking at the detailed drawings it is clear that 274 is not a data, but a register address. Data is actually numbered as 252. Further, Baker appears to disclose a method where part of the address field is used to specify the bits to be loaded. However, Appellant is unable to locate where Baker teaches receiving a data value of a write directed to a control register.

In column 3, lines 15-27, Baker discloses a method and system for writing a plurality of data register bits. The method and system include storing both designation address bits in an address field for addressing a predetermined data register. The designation address bit designates predetermined bits within the data to which the data is to be written as active and other bits within the data register as inactive. Thus Baker includes writing data only to the predetermined bits using single write command. However, Appellant is unable to locate where Baker teaches

interpreting bits of the data value as a data field and bits of the data value as an enable field as required by Appellant's claim 12.

Further, in column 3/lines 65 – column 4/lines 5, column 5/lines 28-29 and column 1/lines 19-41, Runaldue discloses that the bit enable decoder logic 18 may output a unique bit enable signal to each column 14 to enable writing to an addressed word on a bit-by-bit basis. Or the decoder logic 18 may output the same bit enable signal to a selected group of columns to provide variable-width memories, for example converting an 8-bit wide RAM to two 4-bit memory array, or 2-bit memory array.

Runaldue, also discloses that the decoder 62 can be configured to configure the RAM as a full 8-bit wide RAM if mask equals 1 to assert all the bit enable signals for all the columns. Thus, Runaldue appears to disclose the low level details of a random access memory (RAM) array which can be written at the bit level. Runaldue neither discloses anything of the bus level protocol (that is bit enable enclosed in the data phase) nor a system in which control register are written.

Further, Runaldue discloses that memory architecture are typically configured to store a prescribed number of words having a predetermined word length. Such memory structures, known as word organized arrays, have a prescribed number of columns and rows. However, Appellant is unable to locate where Runaldue teaches interpreting bits of the data value as a data field and as enable bits in bit enable field and the number of bits being equal to the number of bits in the control register, as required by the Appellant's claim 12.

As is well established, a prima facie showing of obviousness may only be established if there is a clear suggestion from or in the prior art to make the

modifications proposed by the Examiner. *See Gillette Co. v. S.C. Johnson & Son, Inc.* 919 F. 2d 720 (Fed Cir. 1990).

In the Final Office Action, the Examiner appears to attempt to combine the teachings of Runaldue to write data to an address word on a bit by bit basis using a single write cycle with the teachings of Baker to write to a plurality of data register bits using single write operation by storing both designated address bits and an address field. Appellant submit that the combination does not arrive at the Appellant's invention of claim 12 as the combination does not teach, receiving a data value of a write directed to a control register, interpreting bits of the data value as a data field and as enable bits in bit enable field and the number of bits being equal to the number of bits in the control register, as required by the Appellant's claim 12.

Appellant submits that, in the light of the above arguments, there appears to be no suggestion or motivation for one skilled in the art to combine the references. Therefore, a prima facie case of obviousness in regard to claim 12 has not been established. Appellant respectfully requests the rejection of claim 12 be reversed.

Claims 13-14

Claims 13-14 include claim 12 as a base claim. Accordingly, claims 13-14 are allowable for at least the reasons stated above in regard to claim 12. Appellant believes the above is sufficient to overcome the present rejection of claims 13-14 under Baker and Runaldue. Applicant respectfully requests that the rejection of claims 13-14 be reversed.

Claim 15

Claim15 include claim 12 as a base claim. Accordingly, claim15 are allowable for at least the reasons stated above in regard to claim 12. Moreover, claim 15

requires the processor subsystem posts an entire command sequence in the controller for setting up the IDE (integrated drive electronics) data transfer. Baker, column 14, lines 64-66, discloses that a check is made to determine whether the command is receive, transmit, PCI to/from local bus or auxiliary command.

Also, Runaldue does not appear to teach that the processor subsystem posts an entire command sequence in the controller for setting up the IDE (integrated drive electronics) data transfer, as required by the Applicant's claim 15.

However, Appellant is unable to locate where Runaldue or Baker teaches the processor subsystem posts an entire command sequence in the controller for setting up the IDE (integrated drive electronics) data transfer. Appellant respectfully requests the rejection of claim 15 be reversed.

Claim 16

Claim 16 requires overwriting only the bits at the bit locations of the register for which a corresponding enable bit in the data value is set with corresponding data bits in the data value. The office action has made applicable the rational of rejection regarding claim 20 (as claim 1 has been cancelled) applicable to the claim 16. Appellant therefore submits that the arguments submitted herein above in regard to claims 20 appears to be relevant to claim 16.

Further, Appellant is unable to locate where Baker teaches overwriting only the bits at the bit locations of the register for which a corresponding enable bit in the data value is set with corresponding data bits in the data value.

Also, Runaldue teaches an arrangement for enabling data to be written in an addressed word in memory on bit-by-bit basis, but does not teach overwriting only

the bits at the bit locations of the register for which a corresponding enable bit in the data value is set with corresponding data bits in the data value.

Appellant submits that neither Baker nor Runaldue teach overwriting only the bits at the bit locations of the register for which a corresponding enable bit in the data value is set with corresponding data bits in the data value. The proposed combination does not arrive at the invention of Appellant's claim 16. Appellant respectfully requests the rejection of claim 16 be reversed.

Claims 17-19

Claims 17-19 include claim 16 as a base claim. Accordingly, claims 17-19 are allowable for at least the reasons stated above in regard to claim 16. Further, the office action has made applicable the rational of rejection regarding claims 2-3 and 9-10 applicable to the claims 17-19. Appellant therefore submits that the arguments submitted herein above in regard to claims 2-3 and 9-10 appears to be relevant to the claims 17-19. Appellant respectfully requests the rejection of claims 17-19 be reversed.

Claims 22-23

Claims 22-23 include claim 20 as a base claim. Accordingly, claims 22-23 are allowable for at least the reasons stated above in regard to claim 20. Moreover, claims 22-23 require issuing a write command of the data to the location in memory space for the register.

In column 1/lines 14-41, Runaldue discloses that memory architectures are typically configured to store a prescribed number of words having a predetermined word length. Such memory structures are known as word recognized array and

have a prescribed number of columns and rows. Address decoder size may be reduced using two-level decoding, wherein one level corresponds to a physical word consisting and another level to a logical word.

In the abstract, Baker discloses that register write circuitry 250 writes to a plurality of data register bits 286 using a single register write operation by storing both designated address bits 276 and an address field 274. However, Appellant is unable to locate where Runaldue or Baker teaches issuing a write command of the data to the location in memory space for the register.

Appellant submits that neither Baker nor Runaldue teach issuing a write command of the data to the location in memory space for the register and therefore the combination does not arrive at the Appellant's invention of claim 22-23. Appellant respectfully requests the rejection of claims 22-23 be reversed.

Claims 24-25

Claims 24-25 include claim 16 as a base claim. Accordingly, claims 24-25 are allowable for at least the reasons stated above in regard to claim 16. Moreover, claims 24-25 require issuing a write command of the data to the location in memory space for the register.

Appellant submits that the arguments submitted herein above in regard to claims 22-23 appears to be relevant to the claims 24-25. Appellant respectfully requests the rejection of claims 24-25 be reversed.

CONCLUSION

In view of the foregoing, favorable reconsideration and reversal of the rejections is respectfully requested. Early notification of the same is earnestly solicited. If there are any questions regarding the present application, the Examiner and/or the Board is invited to contact the undersigned attorney at the telephone number listed below.

Respectfully submitted,

September 27, 2006
Date

Paul A. Mendonsa
Reg. No. 42,879
(503) 439-8778

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, LLP
12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026
(503) 439-8778

(viii) Claims appendix.

What is claimed is:

1. (Canceled).

2. (Previously presented) The method recited in claim 20, wherein the register is a control register for a data transfer operation.

3. (Previously presented) The method recited in claim 2, wherein the data transfer operation transfers data to or from a storage device.

4. (Canceled).

5. (Original) The method recited in claim 3, wherein the control register is a command register.

6. (Previously presented) The method recited in claim 20, wherein some of the bits of said register are not overwritten.

7. (Previously presented) The method recited in claim 20, wherein the data field and the bit enable field are received simultaneously.

8. (Original) The method recited in claim 7, wherein the data field is provided at an address which is contiguous with the address for the bit enable field.

9. (Previously presented) The method recited in claim 20, wherein the data transfer operation comprises a data transfer between a processor subsystem and an external storage device or peripheral.

10. (Previously presented) The method recited in claim 9, wherein the processor subsystem posts an entire command sequence for setting up the data transfer.

11. (Previously presented) The method recited in claim 9, wherein the method is carried out in a controller in a bridge connected between the processor subsystem and the external storage device or peripheral.

12. (Previously presented) A computer comprising:

a processor subsystem;

a device which transfers data to or from said processor subsystem; and

a controller connected between said device and said processor subsystem and adapted to control the transfer of data between said device and said processor subsystem, said controller executing a method comprising,

receiving a data value of a write directed to a control register in the controller,

interpreting bits of the data value as a data field, the number of bits in the data field being equal to the number of bits in the control register in the controller and bit locations in the data field corresponding respectively to bit locations in the control register;

interpreting bits of the data value as enable bits in a bit enable field, the number of enable bits in the bit enable field being equal to the number of bits in the control register and bit locations in the bit enable field corresponding respectively to bit locations in the control register; and

overwriting only the bits at the bit locations of the control register for which the enable bit in the corresponding location in the bit enable field is set with the bit in the corresponding location in the data field.

13. (Original) The computer recited in claim 12, further comprising a bridge between the processor subsystem and at least said device, the controller being included in the bridge.

14. (Previously presented) The computer recited in claim 13, wherein the device comprises an IDE (integrated drive electronics) storage device and the bridge comprises an I/O controller hub (ICH) which controls an IDE data transfer between the processor subsystem and the IDE storage device.

15. (Currently Amended) The computer recited in claim 12, wherein the processor subsystem posts an entire command sequence in the controller for setting up the IDE (integrated drive electronics) data transfer.

16. (Previously presented) A software program stored in a tangible medium, said program, when executed, causing a computer to execute a method of writing individual bits of data to a register, said method comprising:

issuing a write of a data value to the register,

overwriting only bits at bit locations of the register for which a corresponding enable bit in the data value is set with corresponding data bits in the data value.

17. (Original) The software program recited in claim 16, wherein said software program comprises a driver in the operating system software executed by a processor subsystem in the computer.

18. (Previously presented) The software program recited in claim 17, wherein the register is a control register in a controller adapted to control an IDE (integrated drive electronics) data transfer operation between said processor subsystem and an IDE storage device.

19. (Previously presented) The software program recited in claim 17, wherein

the processor subsystem posts an entire command sequence for setting up the IDE (integrated drive electronics) data transfer to the controller.

20. (Previously presented) A method comprising

receiving data of a single write command wherein the data comprises a bit enable field and a data field comprising N bits in each field, and

updating a register with one or more bits of the data field that are associated with enabled bits of the bit enable field.

21. (Previously presented) The method of claim 20 wherein

the data of the single write command comprises 2*N bits,

the bit enable field comprises N bits, and

the data field comprises N bits.

22. (Previously presented) The method of claim 20 wherein the register has a location in configuration space and a location in memory space, further comprising

issuing the single write command of the data to the location in memory space for the register.

23. (Previously presented) The method of claim 20 wherein

the control register has a location in I/O space and a location in memory space, and

the processor subsystem issues a write of the data value to the location in memory space for the register.

24. (Previously presented) The method of claim 16 wherein

the data value comprises N enable bits and N data bits that correspond to N bits of the register.

25. (Previously presented) The method of claim 16 wherein

the register has a location in configuration space and a location in memory space, and

issuing a write of a data value to the register comprises issuing the write to the location in memory space for the register.

(ix) Evidence appendix.

None.

(x) Related proceedings appendix.

None.